

P0492002 0016 US1

1/30

HARE ET AL

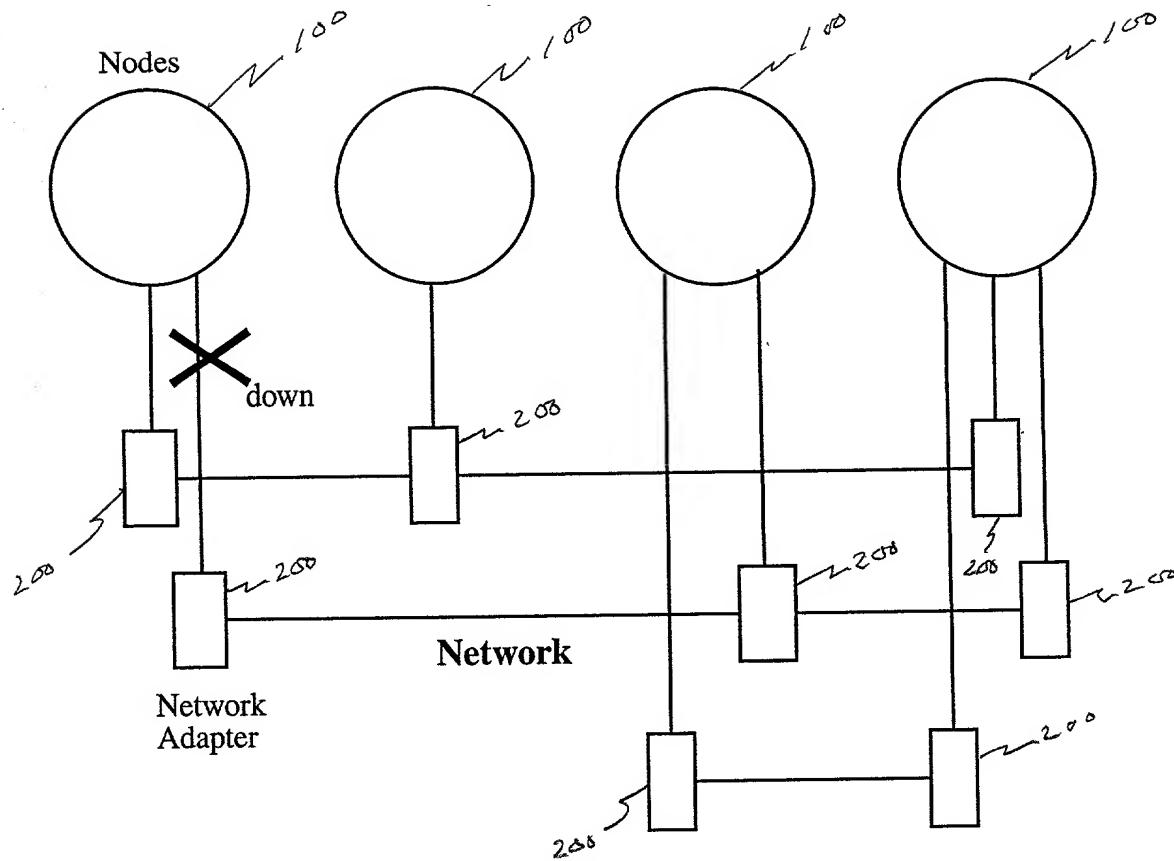


Figure 1: Nodes and physical connectivity

2/30

PON 920020016 US1

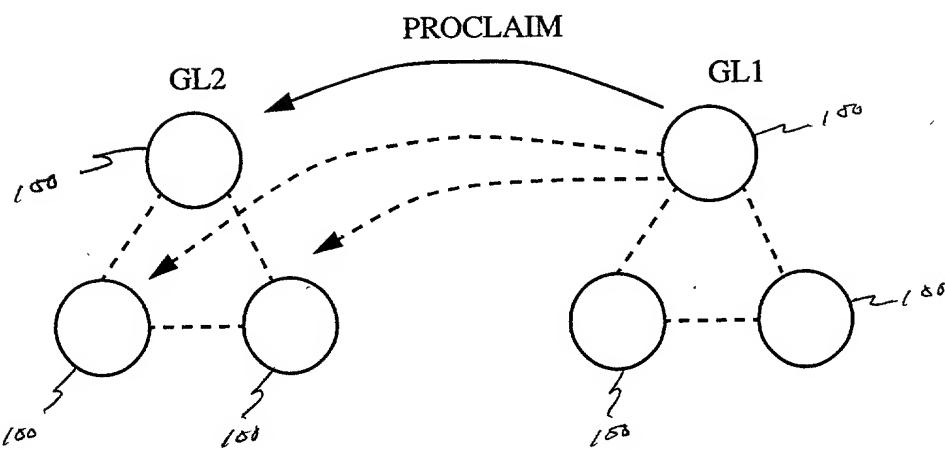


Figure 2a) JOIN Protocol: PROCLAIM message

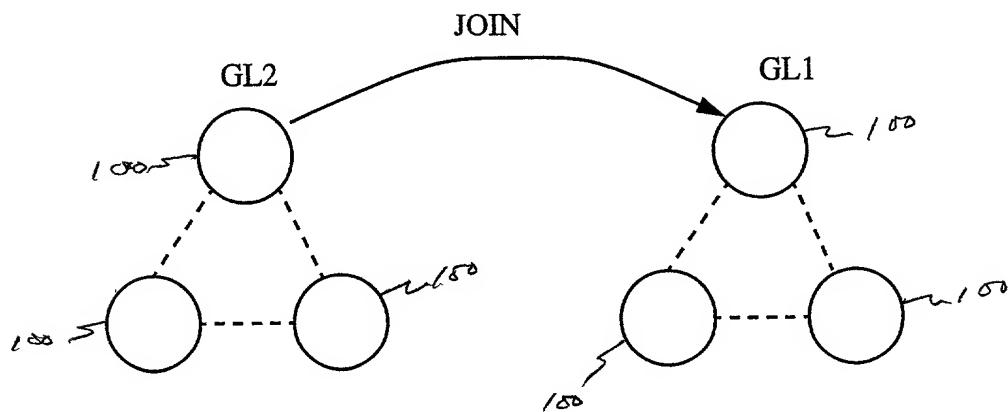


Figure 2b) JOIN Protocol: JOIN message

3/30

Pou 920020016451

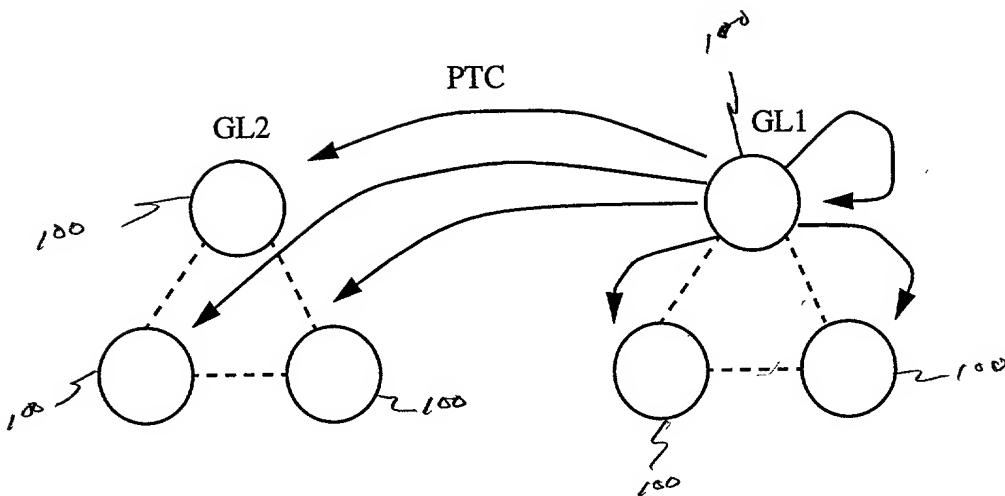


Figure 2c) JOIN Protocol: PTC message

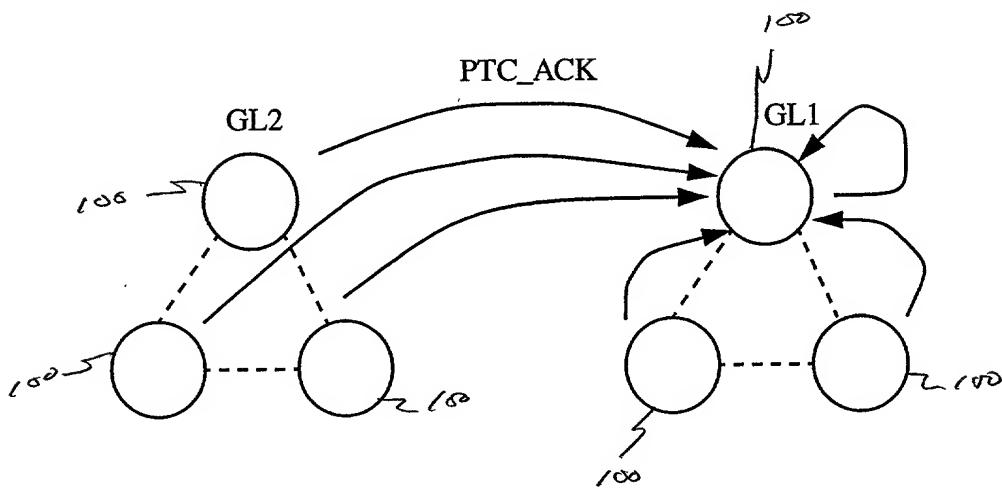


Figure 2d) JOIN Protocol: PTC_ACK message

4/30

Pou 92002 0016 US1

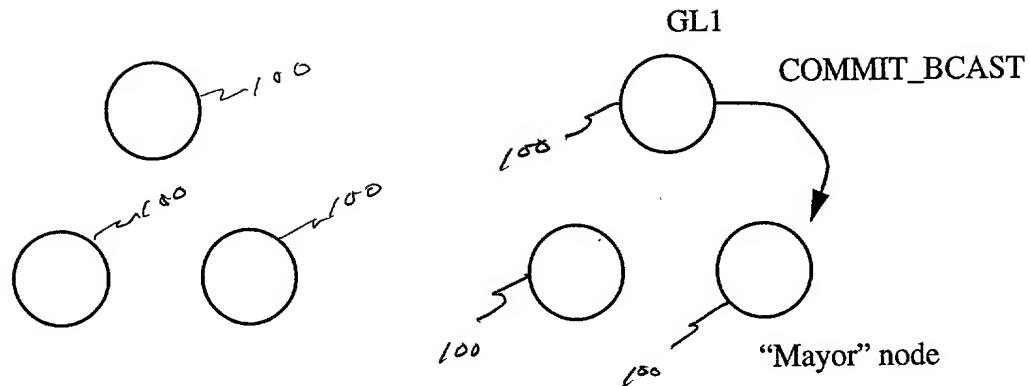


Figure 2e) JOIN Protocol: COMMIT_BCAST message

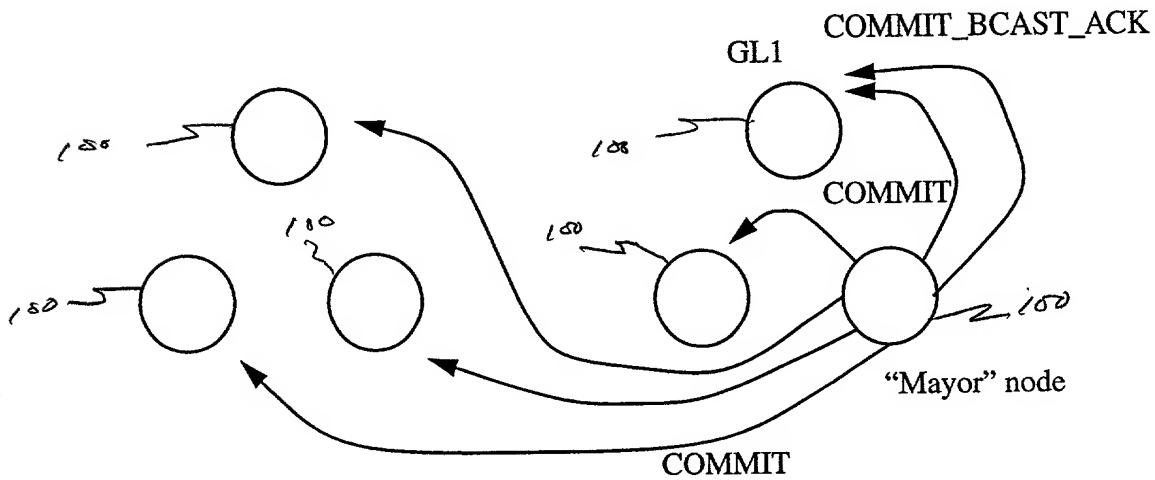


Figure 2f) JOIN Protocol: COMMIT and COMMIT_BCAST_ACK messages

5/30

Pou9 2002 0016451

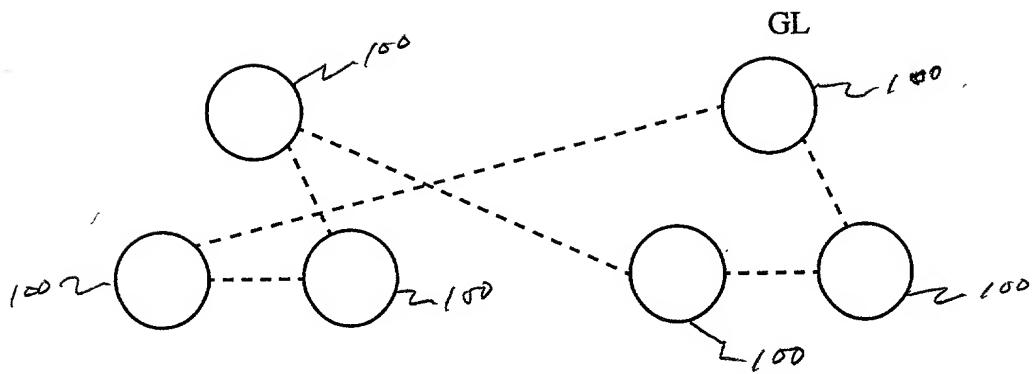


Figure 2g) JOIN Protocol: new group formed after completion of protocol

Figure 2: JOIN protocol

6/30

Pou 920020016 US1

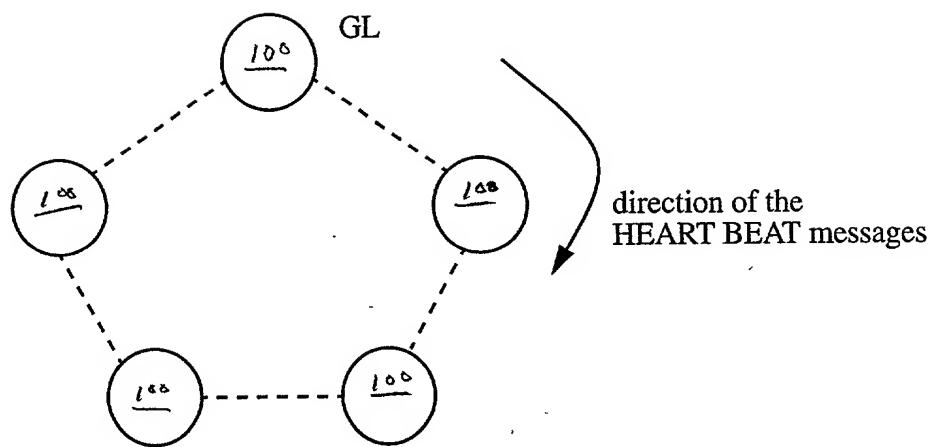


Figure 3a) DEATH Protocol: initial state: heartbeat ring

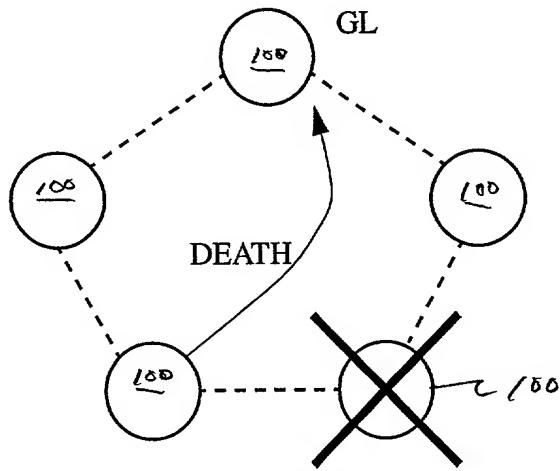


Figure 3b) DEATH Protocol: DEATH message

7/30

Pou 92002 0016451

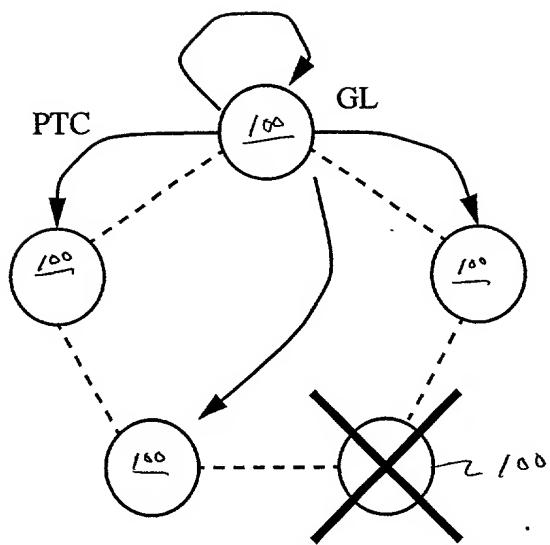


Figure 3c) DEATH Protocol: PTC message

Figure 3: DEATH protocol

3.001760376 021502

8/30
Pou 920020016451

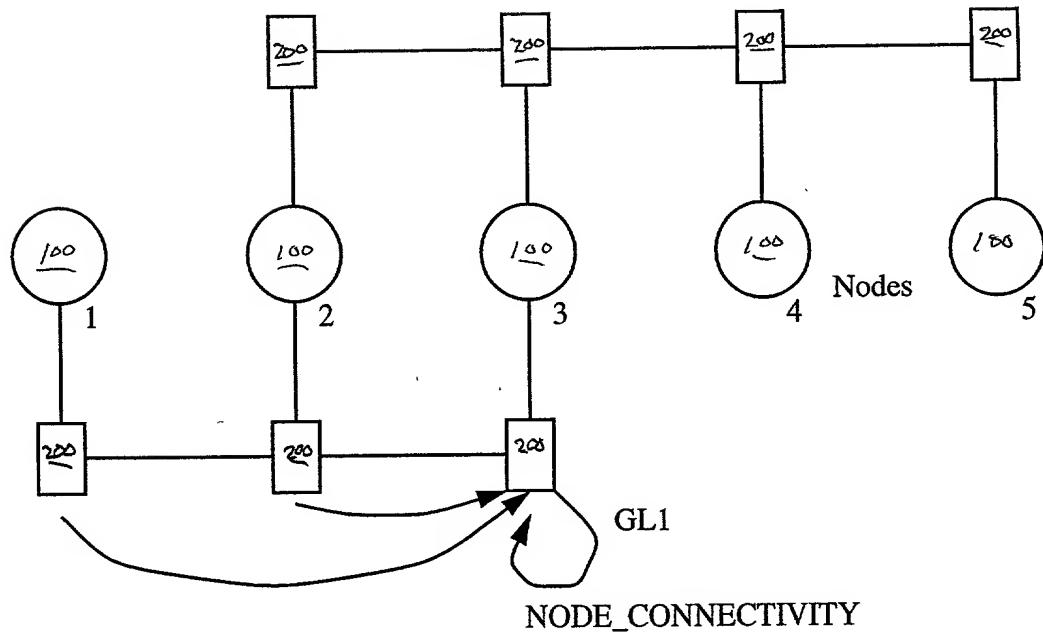


Figure 4a) Node Reachability Protocol: NODE_CONNECTIVITY message

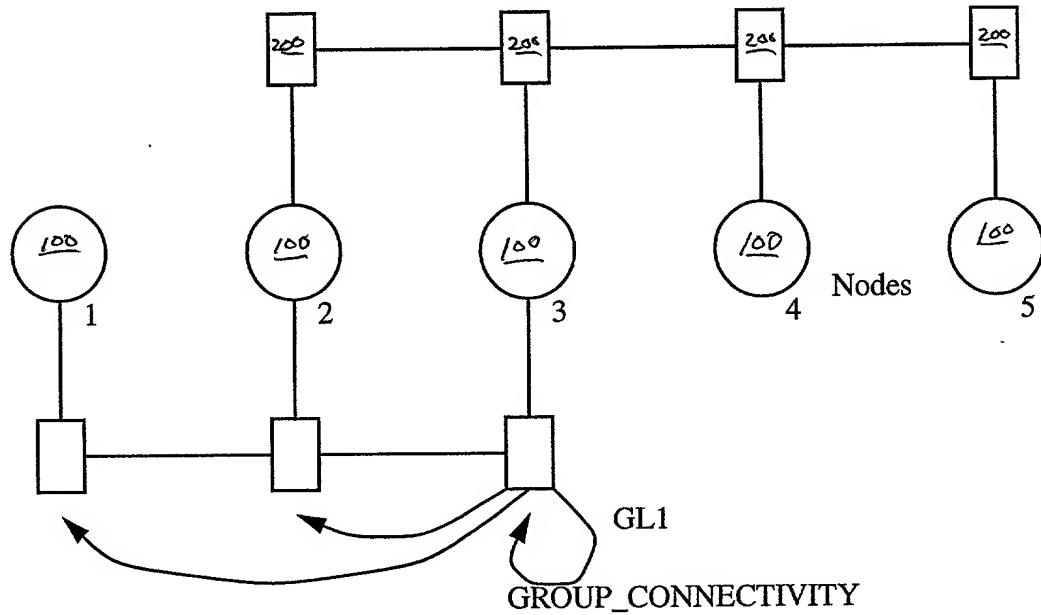


Figure 4b) Node Reachability Protocol: GROUP_CONNECTIVITY message

9/30

P04920020016451

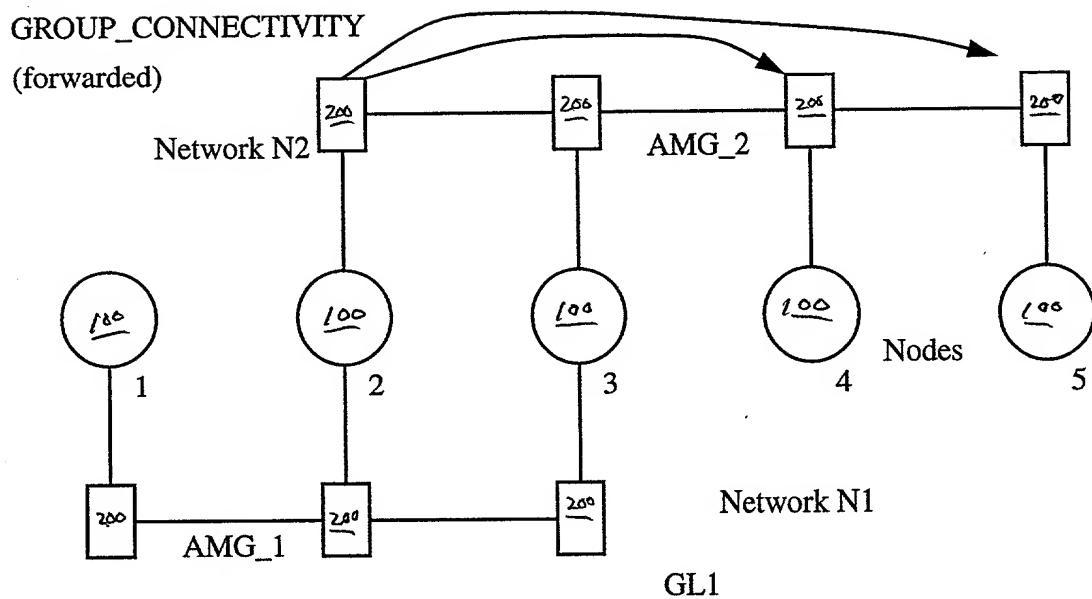
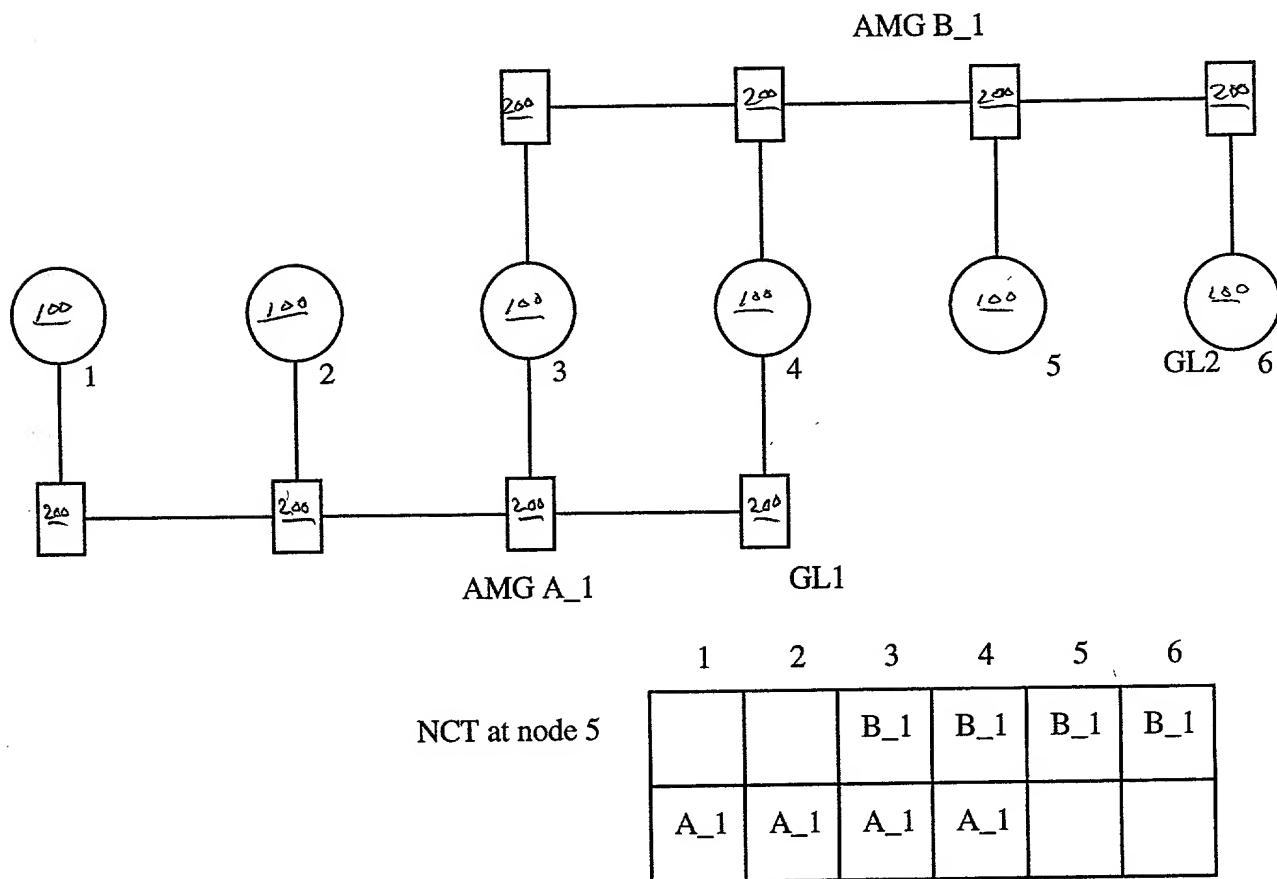


Figure 4c) Node Reachability Protocol: forwarding of GROUP_CONNECTIVITY message

Figure 4: Node reachability protocol: NODE_CONNECTIVITY and GROUP_CONNECTIVITY messages

10/30

P04920020016451



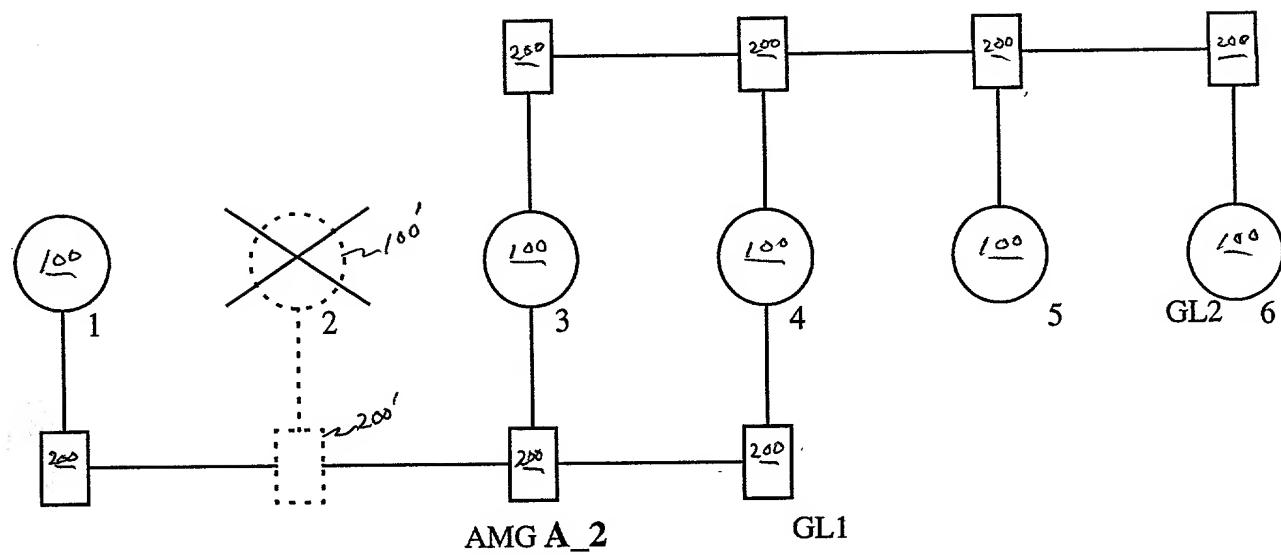
5a) Initial situation

Figure 5: Topology Propagation Scenario: node death

11/30

P049200200164S1

AMG B_1

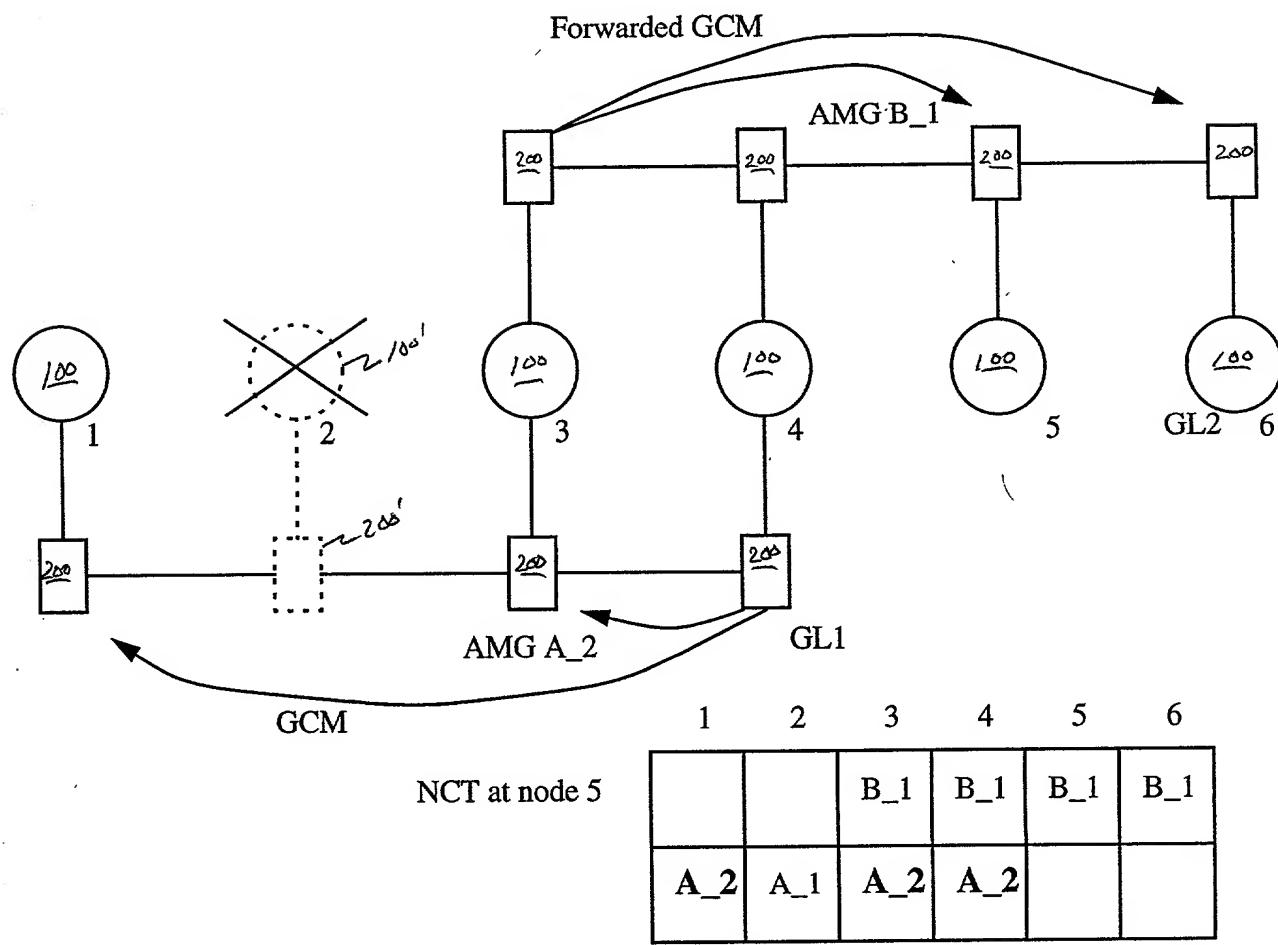


1	2	3	4	5	6
		B_1	B_1	B_1	B_1

5b) Node 2 dies: Nodes 1, 3, and 4 form AMG A_2

12/30

Pou920020016451



5c) GCM for AMG A_2 is propagated to all nodes

13/30

Port 92002 0016 US1

Network A: 10 seconds detection time
Network B: 40 seconds detection time

Nodes 1 and 2 are forming AMGs on networks A and B

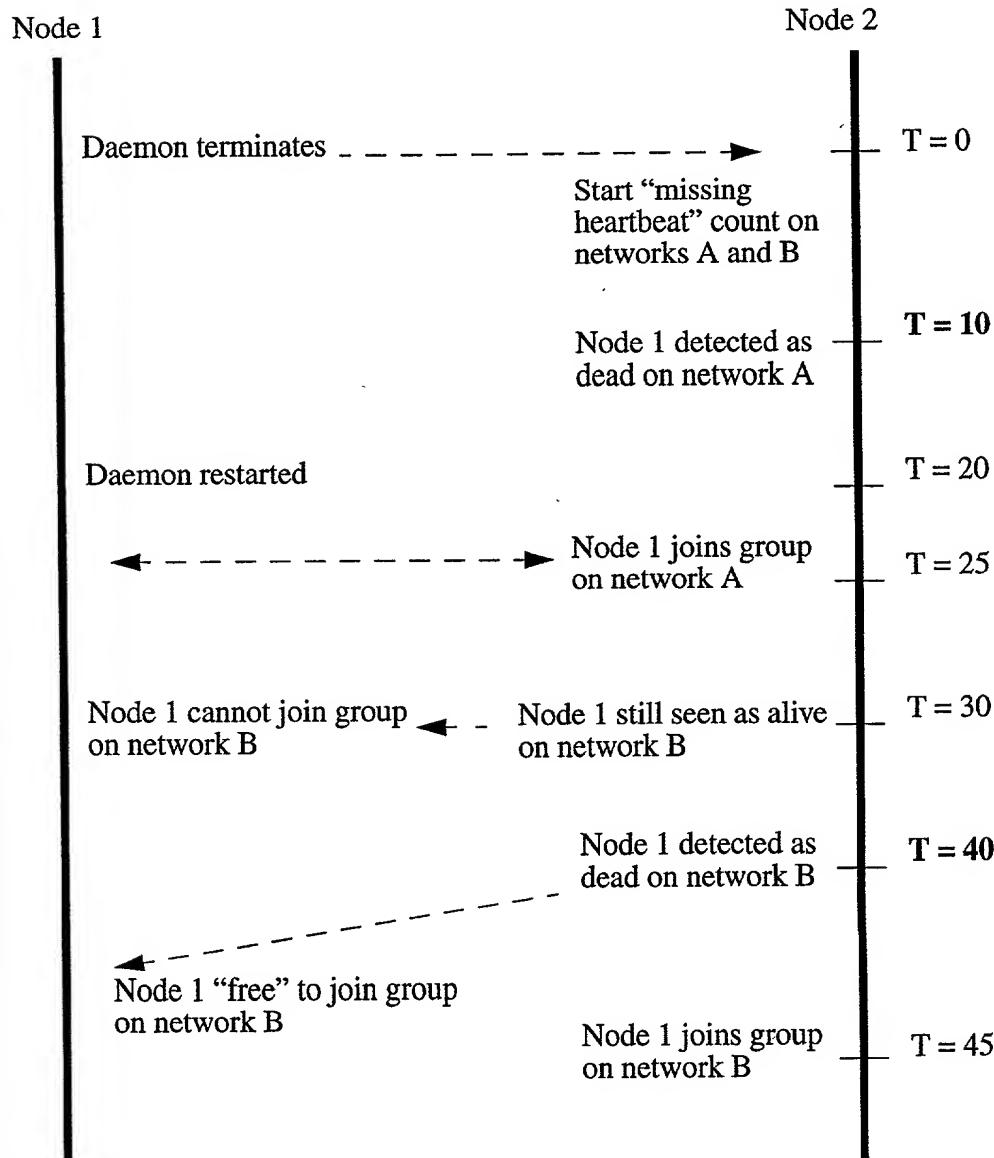


Figure 7) Inconsistency caused by quick daemon restart in the presence of different detection times for each network: the daemon on node 1 goes down and is restarted, but this is never detected by node 2.

14/30

Pou920020016451

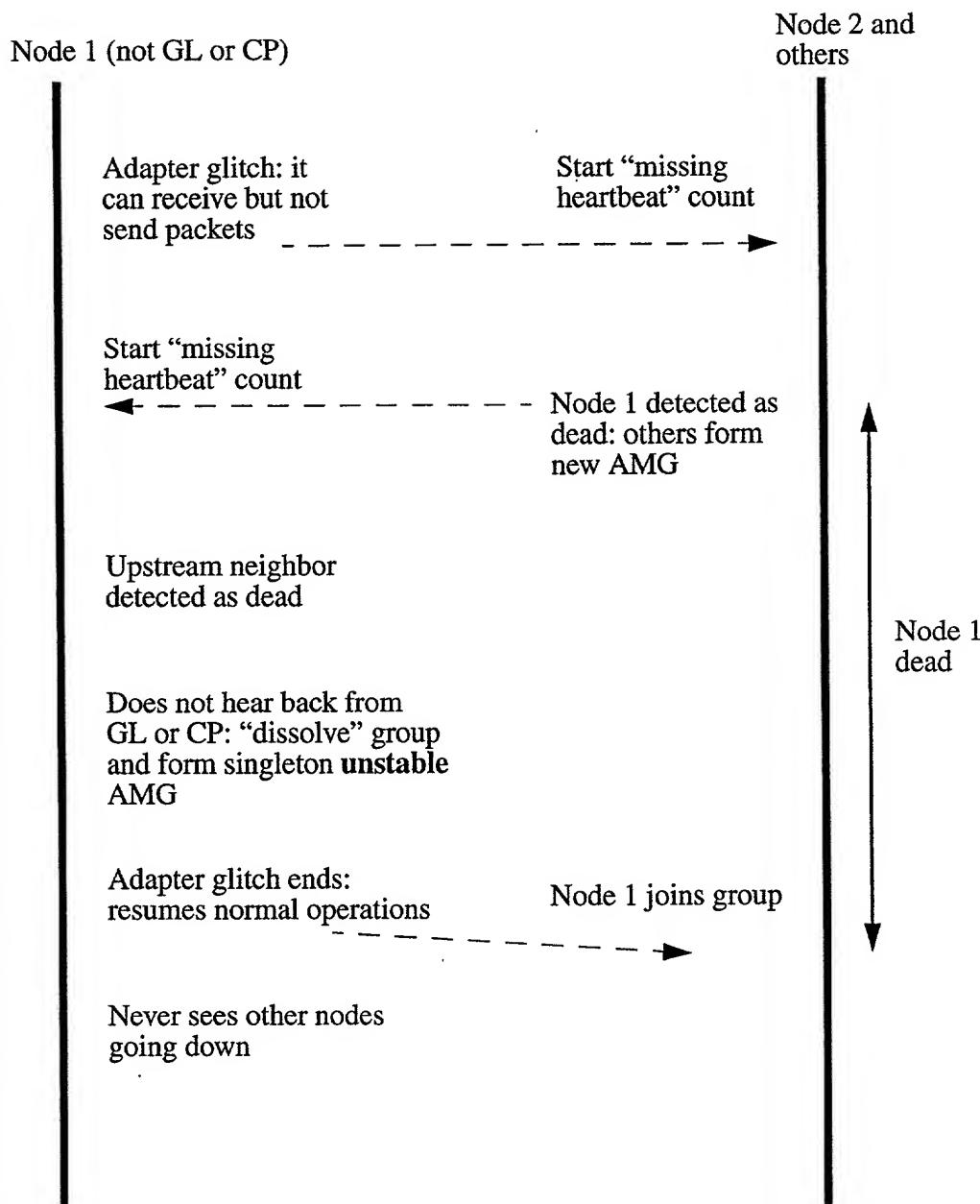


Figure 7) Inconsistency caused by temporary communication problem in an adapter: node 1 is not the GL or CP in its group. Node 1 never notices other nodes as down, though the others do see node 1 being unreachable.

15/30

P049200200162151

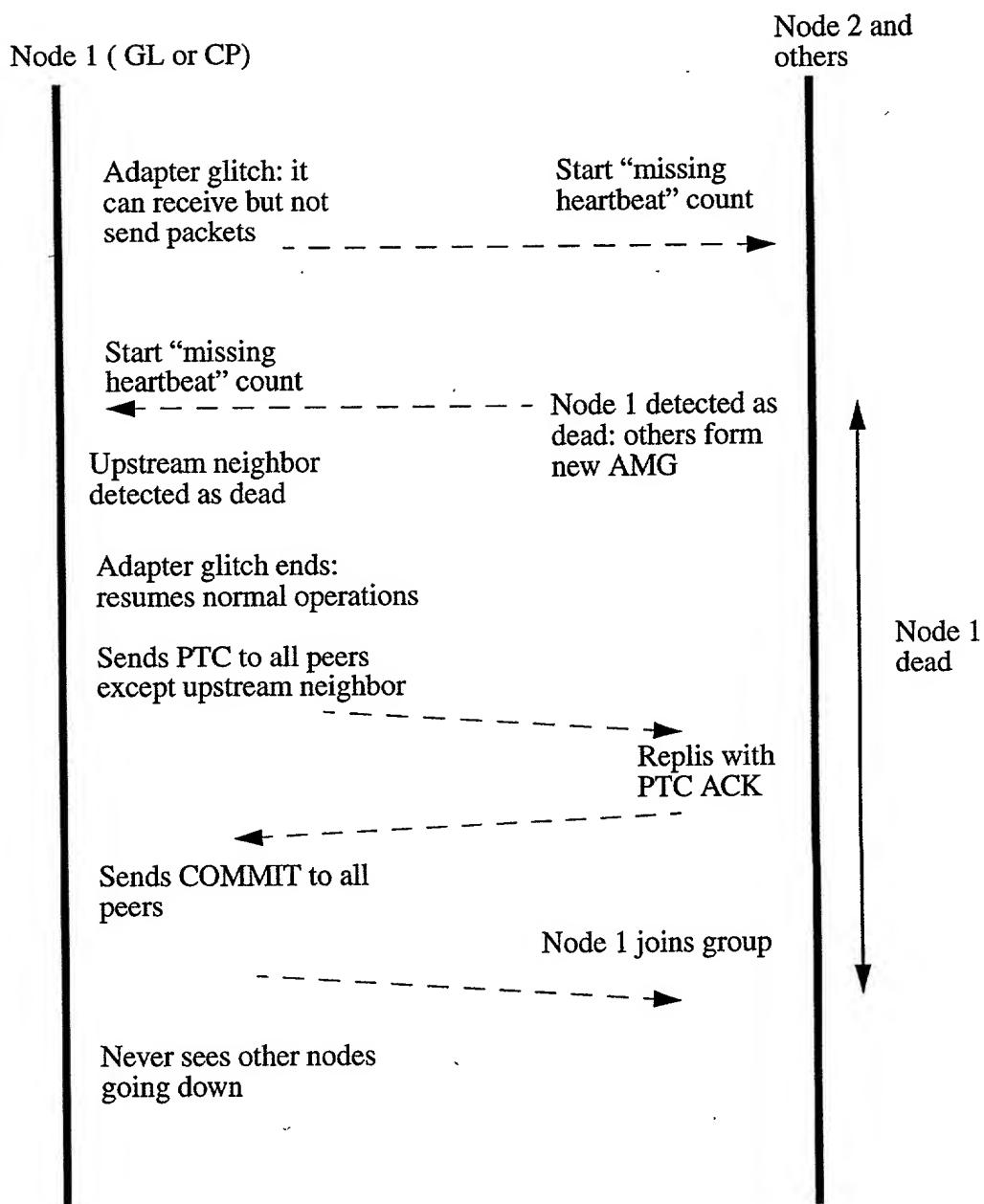


Figure 7) Inconsistency caused by temporary communication problem in an adapter: node 1 is the GL or CP in its group. Node 1 never sees the other nodes as unreachable, while the others do see node 1 as unreachable for a period.

16/30

P0492002 0016451

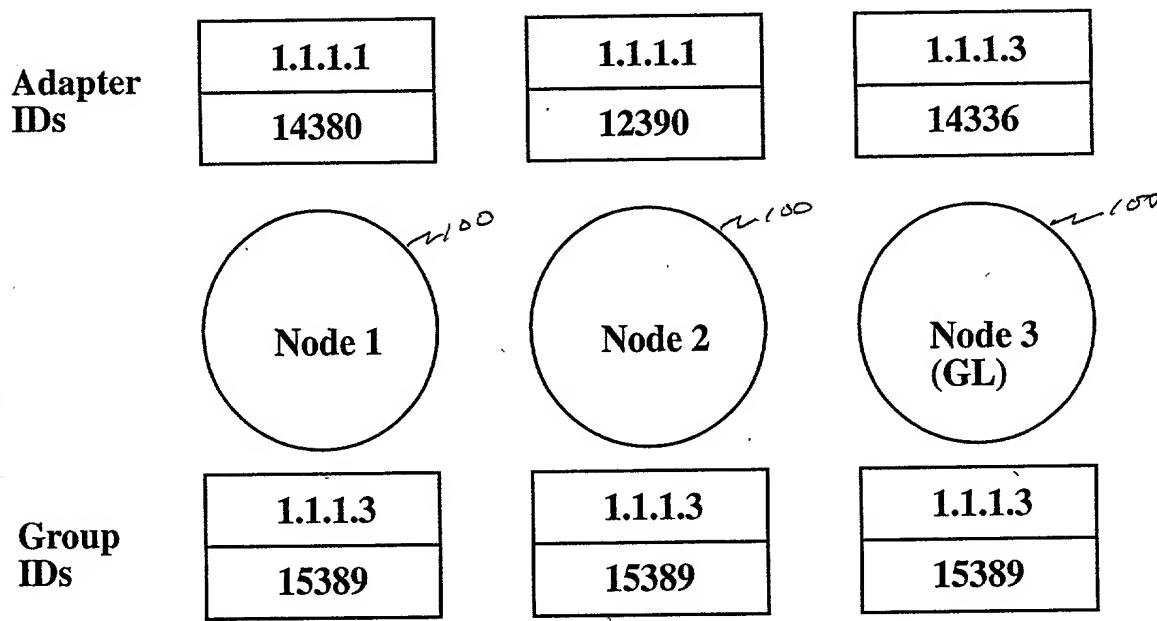


Figure 10) Adapter IDs and Group IDs. An adapter ID has the IP address of the adapter and and instance number. The Group ID has the IP address of the Group Leader and an instance number that changes each time the group changes.

17/30

Pou 92002 0016 451

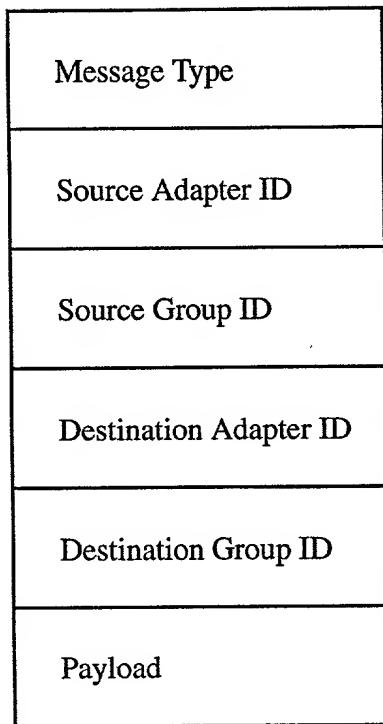


Figure 10
Format of the protocol packets that are sent over the network

18/30

Pou 920020016 US1

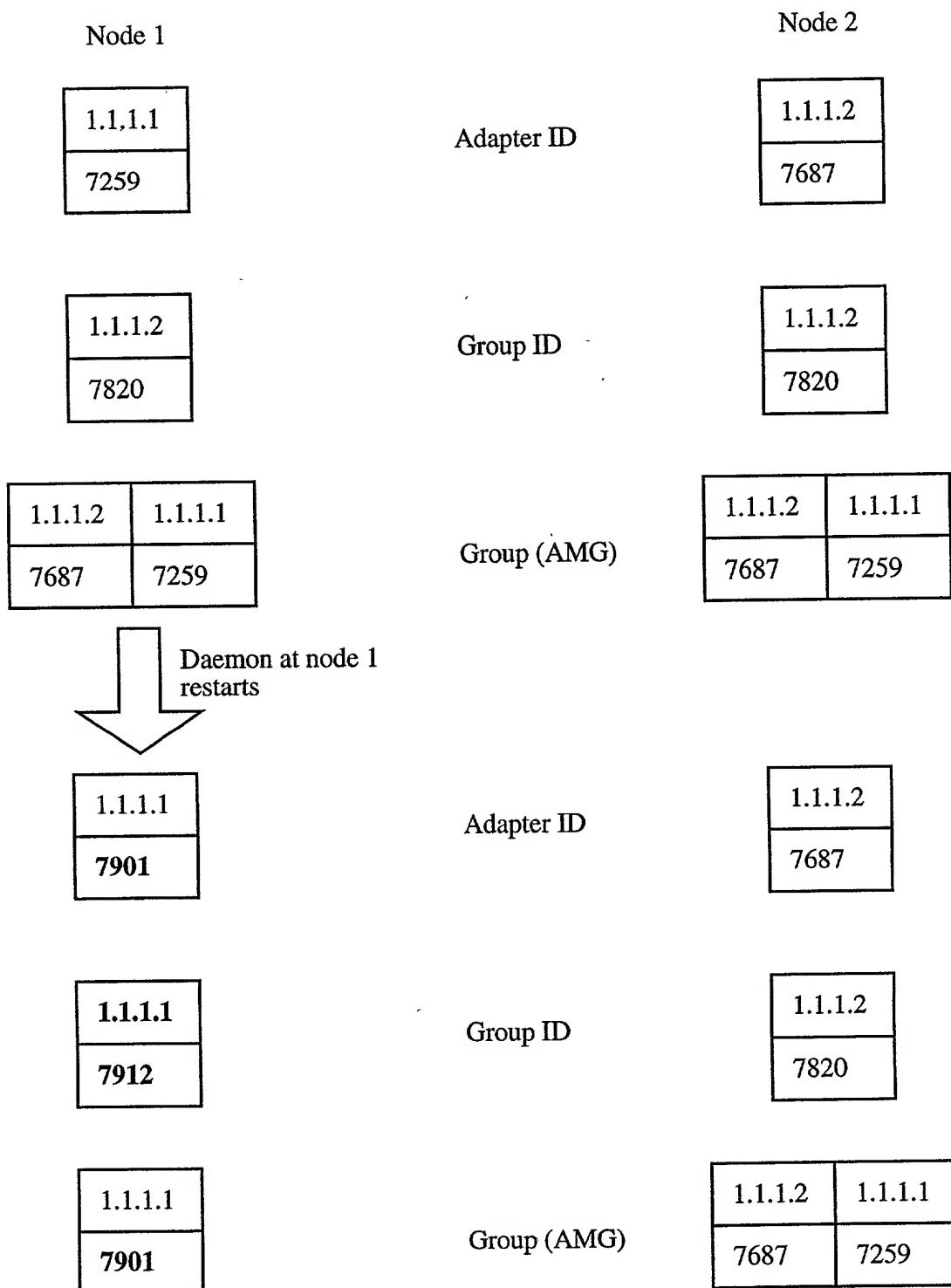


Figure 18 Adapter and Group IDs when the daemon at node 1 terminates and is restarted.

19/30

P0492002 0016451

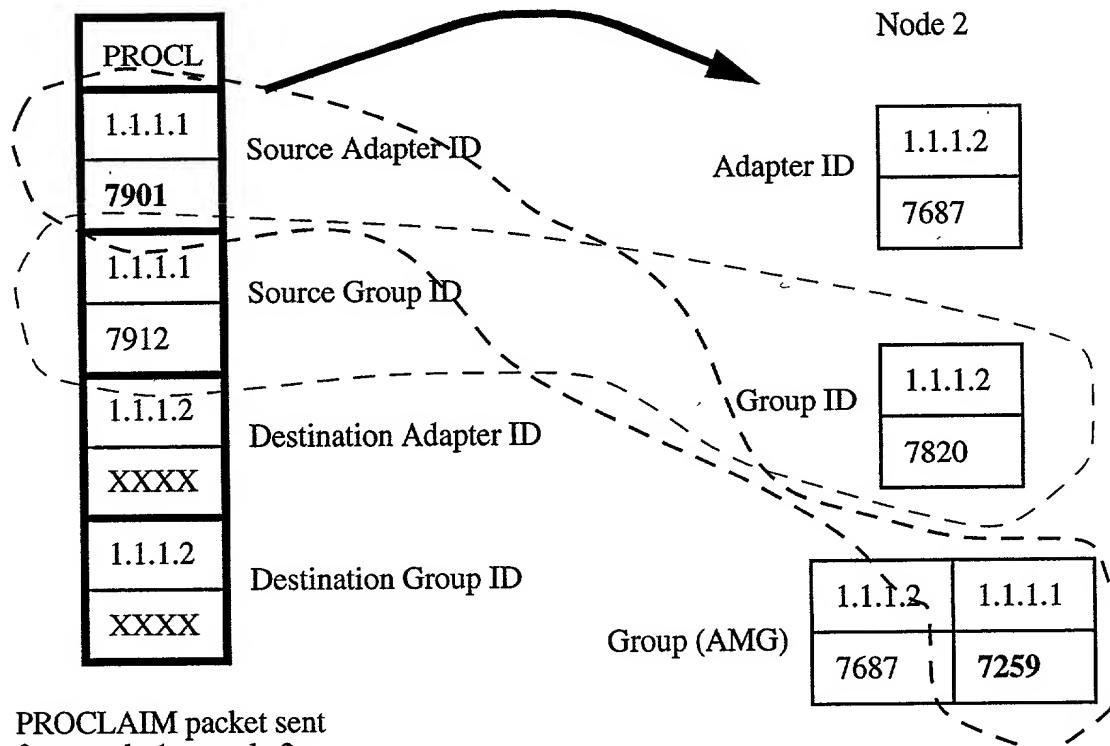
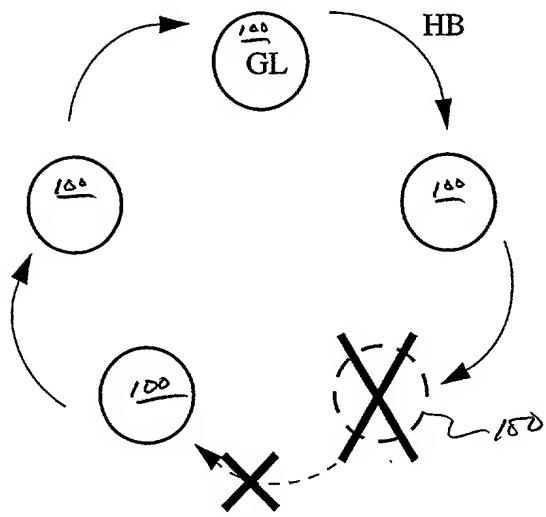
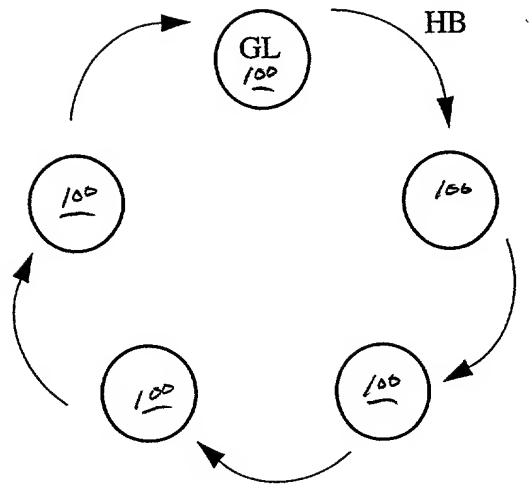


Figure 12) A “live” node detects that a remote daemon restarted. The Group ID of the message is different from node 2’s, while the address of the sender is listed on node 2’s group membership.

20/30

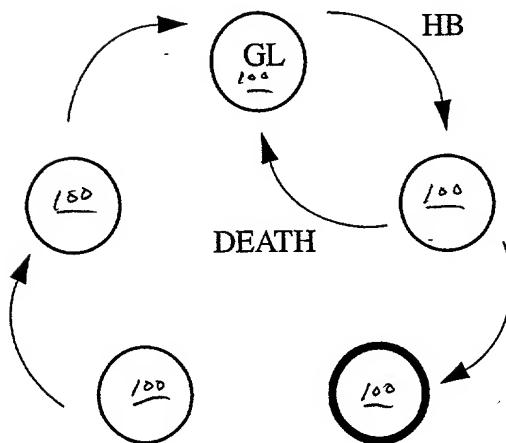
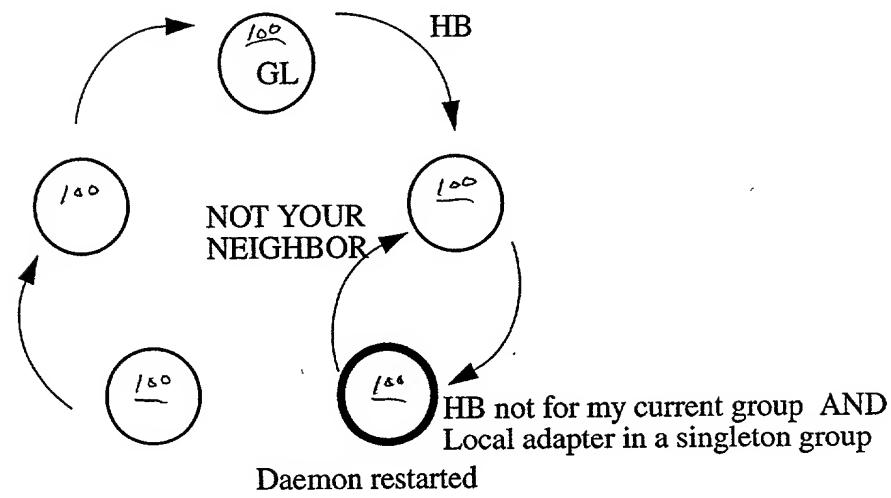
P04920020016 US1



\downarrow^3
Figure 4(a) A daemon that is restarted detects that a previous instance used to belong to an AMG because of heartbeat messages that it receives while in a singleton group.

21/30

P04920020016 US1



13
Figure 2b) Continuation

22/30

Pon 92002 0016 451

Node 3 (GL)	Node 2
1.1.1.3 7687	Adapter ID
1.1.1.3 7820	Group ID
1.1.1.3 7687	Group (AMG)
1.1.1.3 7687	last_stable_group
1.1.1.2 7259	1.1.1.2 7259
1.1.1.1 7228	1.1.1.1 7228

Communication glitch
in node 3's adapter.

↑
Figure 5a) Solution to the Quick Communication Interruption Problem. initial state:
nodes 1,2, and 3 are part of the same AMG., Node 3's adapter suffers a temporary
failure.

23/30

Pon 92002 0016 451

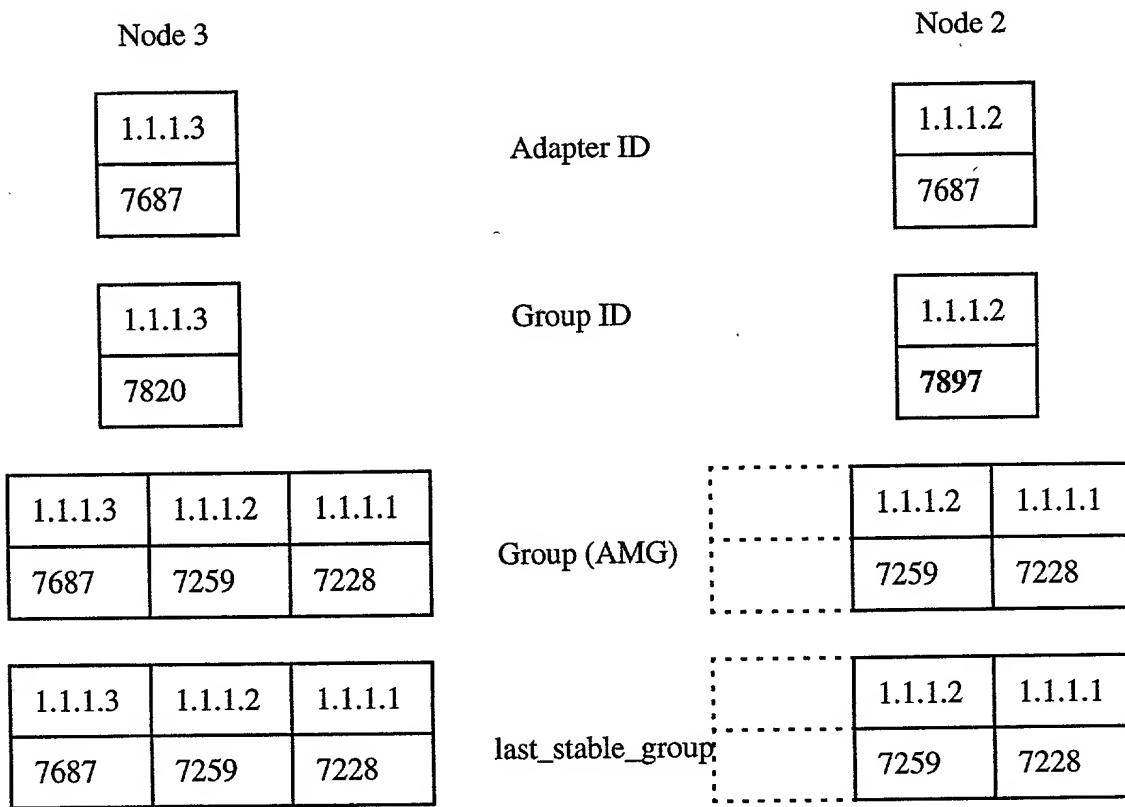


Figure 5b) Solution to the Quick Communication Interruption Problem. Node 3's adapter suffers a temporary failure. Node 2 commits a new AMG, while node 3 is still in the process of missing HBs from its neighbor

24/30

Pou920020016 USI

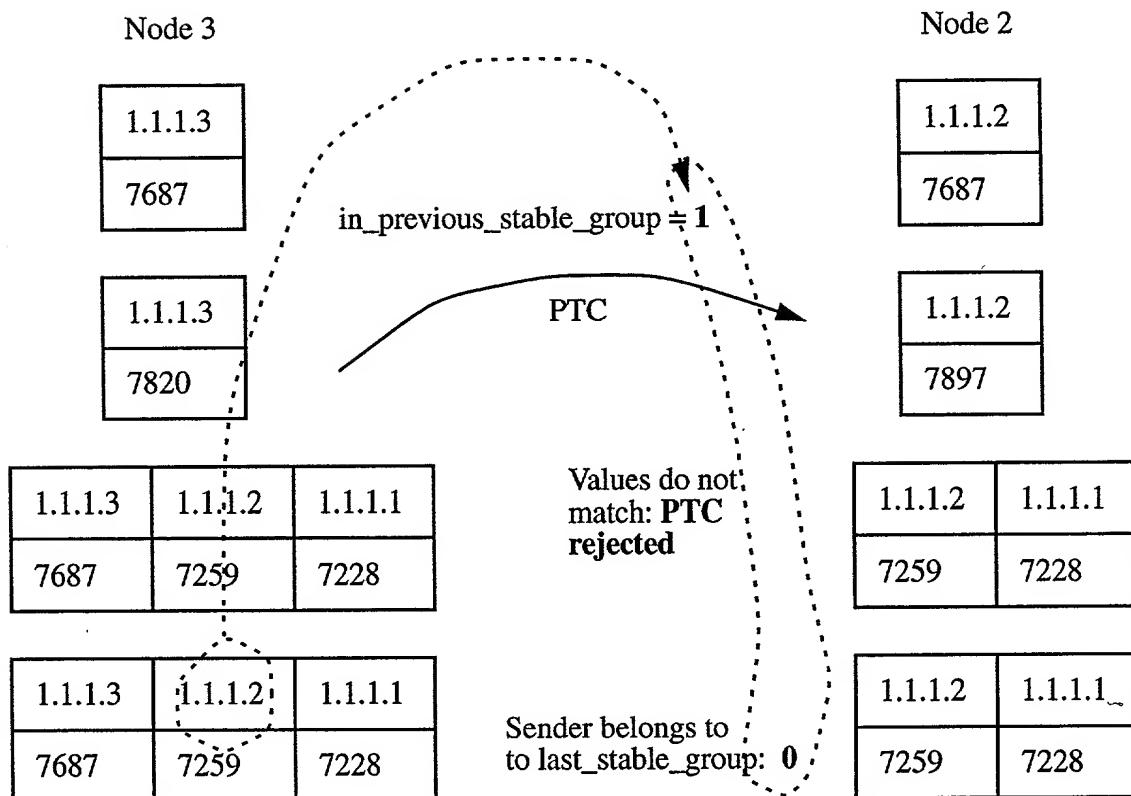


Figure 5c) Solution to the Quick Communication Interruption Problem. Node 3 sends a PTC when it stops receiving HBs from its upstream neighbor. The PTCs are rejected because of the discrepancy in the last_stable_group results.

25/30

Pou920020016 USI

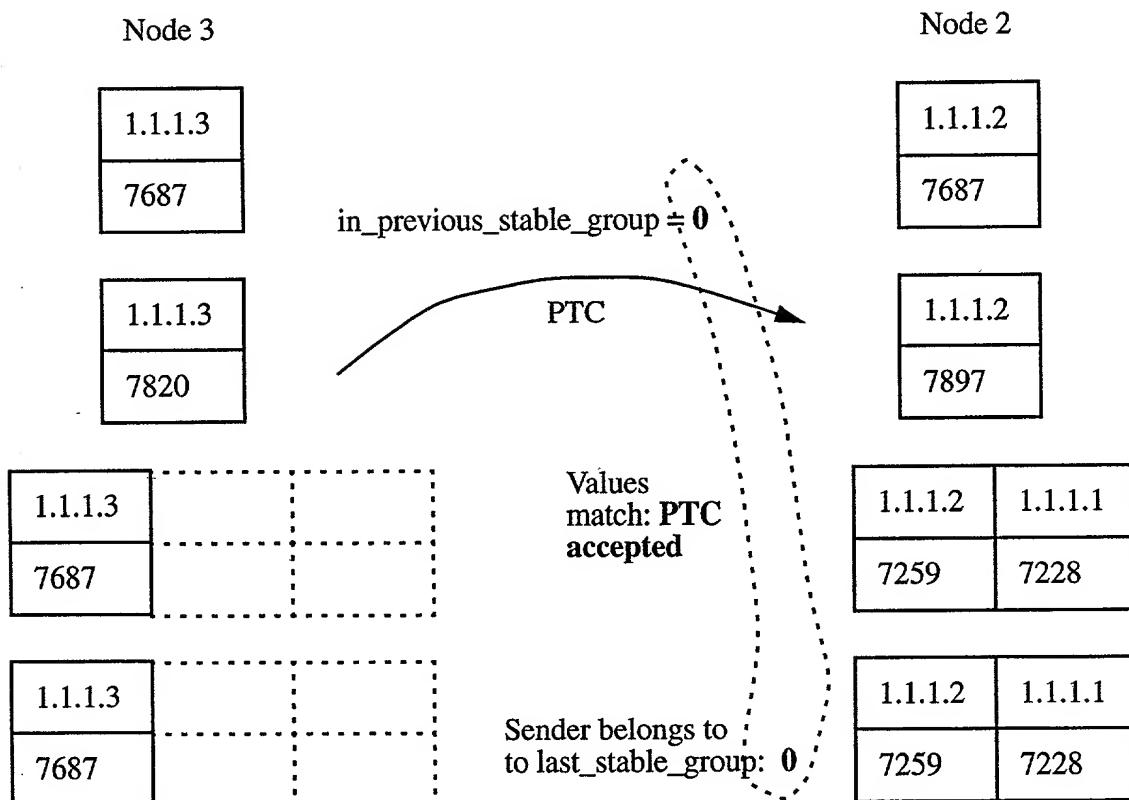


Figure 14d) Solution to the Quick Communication Interruption Problem. Since node 3 does not get replies to its PTC messages, it is forced to form a singleton group. At this point, it updates last_stable_group. From then on node 3's PTC are accepted again.

26/30

Dom 9/2002 0016 USI

Node 1	Adapter ID			Node 3 (GL)
	1.1.1.1			1.1.1.3
	7228			7687
	Group ID			
	1.1.1.3			1.1.1.3
	7820			7820
	Group (AMG)			
	1.1.1.3	1.1.1.2	1.1.1.1	1.1.1.3
	7687	7259	7228	7687
	1.1.1.3	1.1.1.2	1.1.1.1	1.1.1.3
	7687	7259	7228	7259
	last_stable_group			
	1.1.1.3	1.1.1.2	1.1.1.1	1.1.1.3
	7687	7259	7228	7259

Communication glitch
in node 1's adapter.

Figure 5a) Solution to the Quick Communication Interruption Problem. initial state:
nodes 1,2, and 3 are part of the same AMG., Node 1's adapter suffers a temporary
failure.

27/30

Pon 92002 0016 451

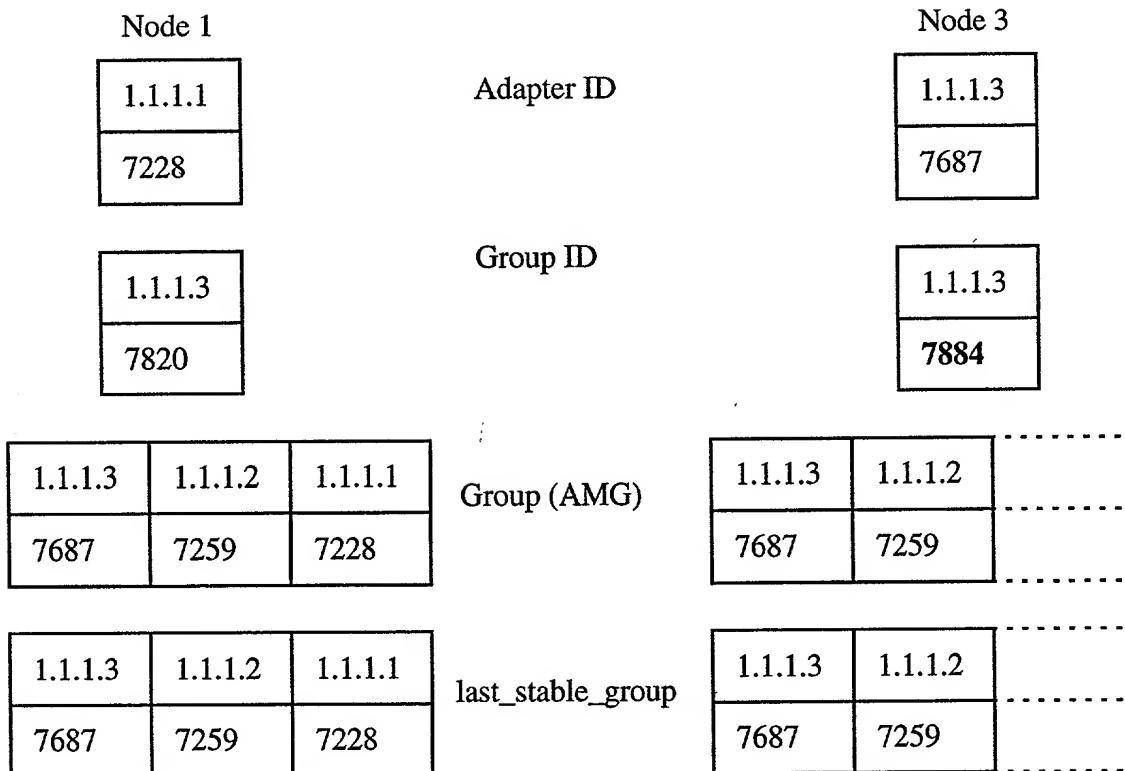


Figure 2b) Solution to the Quick Communication Interruption Problem. Node 1's adapter suffers a temporary failure. Node 3 commits a new AMG, while node 1 is still in the process of missing HBs from its neighbor

28/30

Pou920020016 US1

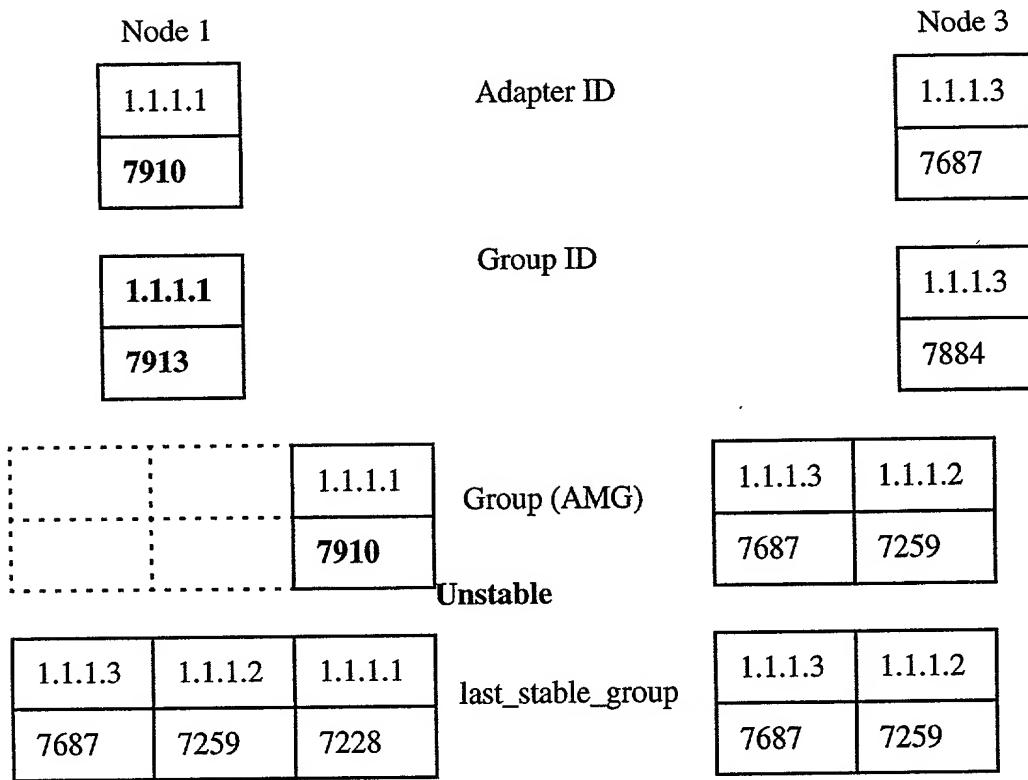


Figure 15) Solution to the Quick Communication Interruption Problem. Node 1 dissolves its group and forms a singleton unstable group. Note that because the group is unstable, there is no change in last_stable_group.

29/30

Pou 9 2002 0016 USI

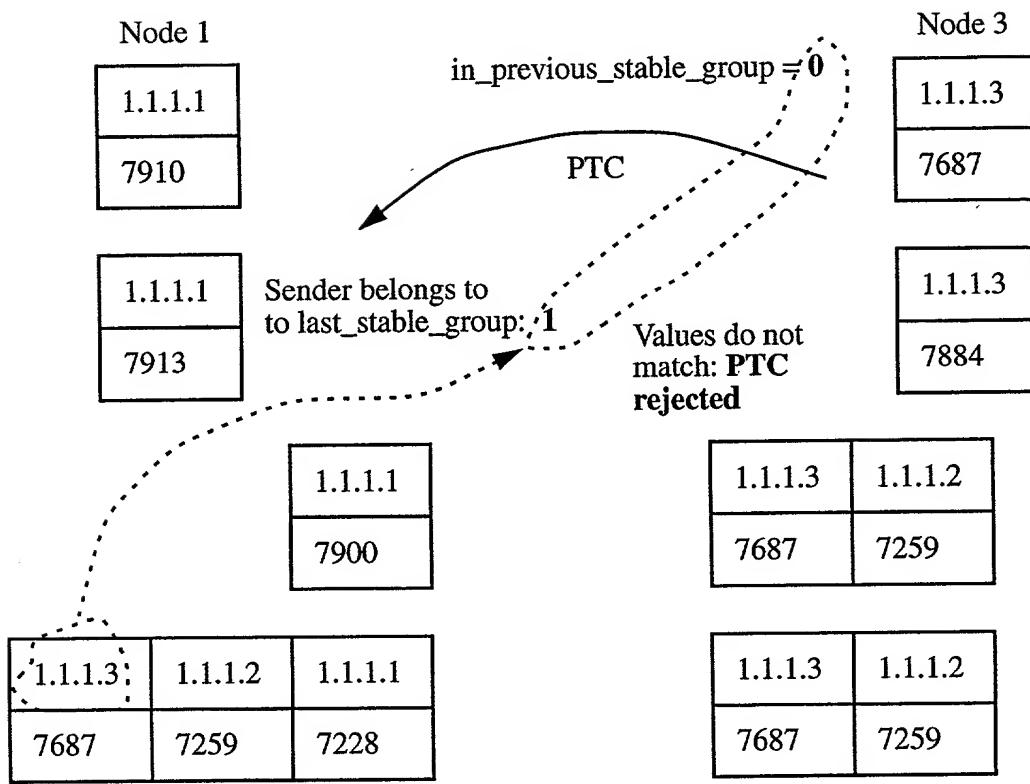


Figure 2d) Solution to the Quick Communication Interruption Problem. Node 3 sends a PTC when node 1 responds the PROCLAIM message with a JOIN. The PTCs are rejected because of the discrepancy in the last_stable_group results.

30/30

Dom 92002 0016 451

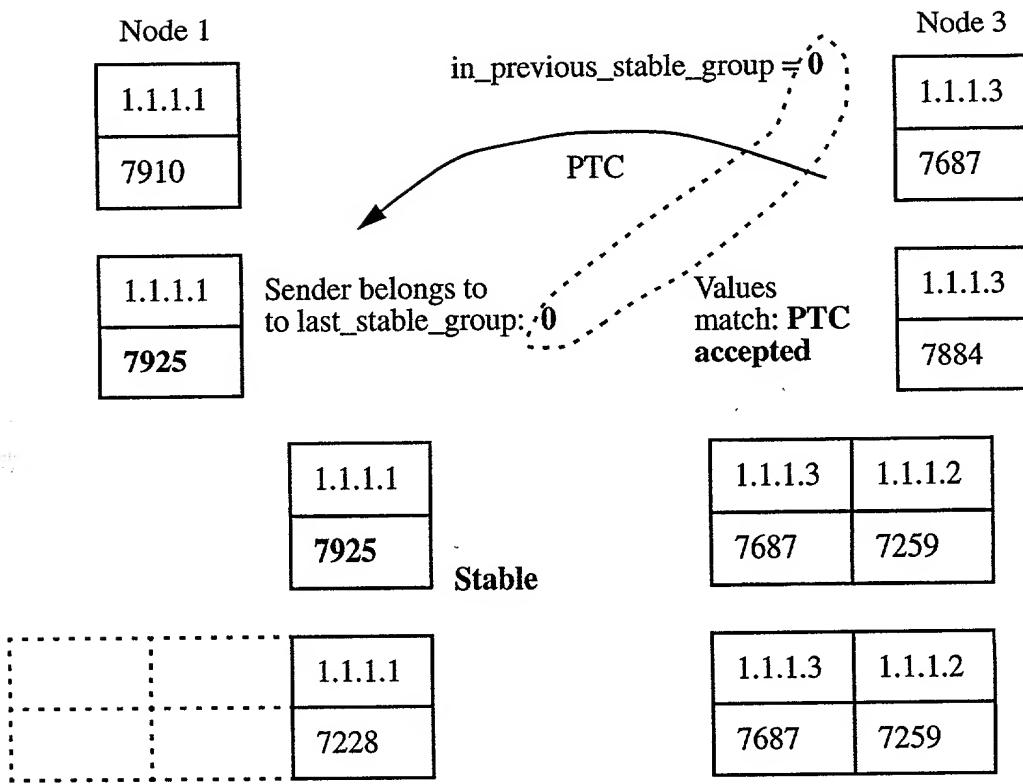


Figure 2e) Solution to the Quick Communication Interruption Problem. Since node 3 does not get replies to its PTC messages, it is eventually forced to form a singleton group. At this point, it updates last_stable_group. From then on node 3's PTC are accepted again.